Phil Springer

Senior AI Platform Engineer

Phoenix, AZ · phil@philspringer.com · (503) 734-9982 · philspringer.com · linkedin.com/in/philspringerai

Summary

I use AI-assisted engineering to build infrastructure and tooling: an agent harness, release gates, MCP server vetting and model routing. My AI work uses planted-bug tests, review by a model from another lab and human approval for sending, spending and deletion. I run local model inference and backups with sampled restore tests. I configured a GitHub Enterprise Cloud lab.

Experience

Founder & Principal Engineer, IM-Empowered Inc. · Phoenix, AZ (Remote) · 2010 to Present

Agentic engineering platform

- AI-assisted engineering. My agents and I built an agent workflow harness. It gates sending, spending, deletion and release, and every change gets a review by a model from another lab. In one week (Nov 10 to 16, 2026) it dispatched 1,612 agent jobs; 1,308 ran to completion and 268 were recorded as failed.
- Infrastructure. I run self-hosted Git, local model inference and backups with sampled restore tests.
- Guardrails. Every safety control I ship gets planted-bug tests and a second-model review. One recent gate passed 35 of 35 tests and caught all 13 bugs we planted before it went live.
- GitHub Enterprise Cloud lab. I set a ruleset requiring a pull request, code-owner review and approval by someone other than the last pusher. With it active, a direct push to main was rejected. The audit log recorded the ruleset's creation and update. For Copilot Business, I set new features to off by default, turned off web search, outside indexing and bring-your-own-key access, blocked public-code matches and turned on metrics.
- Human oversight. Each agent has one job and only the tools for it, and work moves between roles in a fixed order. Sending, spending and deletion require my recorded approval.
- MCP server vetting (toolwitness.ai). My agents and I built toolwitness.ai: it starts every new MCP server in a no-network sandbox and records what it does. From Oct 16 to Nov 17, 2026, it ran 2,479 automated listing reviews: 1,593 published, 812 rejected and 74 that could not be measured. A tool is listed only after scripted checks and independent AI reviews.
- Context management. A gate limits the size of the shared rule file and of each repo's rule file, and a log shows which files each session loaded.
- Live model routing. My router picks the lowest-cost capable model for eligible AI helper tasks.
- Data pipeline. My pipeline pulls job postings daily from public hiring APIs through Python modules with automated tests.
- Grounded output. My job pipeline rejects any answer that cannot quote its source word for word. My retrieval chain answers from source passages.
- Multi-agent system. An MCP server connects the agents to the task tracker.
- Scheduled publishing. I run automated newsletter sites Monday, Wednesday and Friday. Each issue passes automated checks before it is sent.

Email marketing, traffic and conversion

- Owned paid media, list building, deliverability, offers and back-end resale.
- Ran my own email and traffic business.

Network Administrator and Webmaster, Power Organics · Mount Shasta, CA (remote from 2010) · 2000 to 2023

- Sole technical owner for the company's network, servers, website and systems. From 2010, administered servers and supported staff machines remotely.
- Ran the online store: holiday sales, specials and promotions.

Skills

AI: agent orchestration, MCP, Claude Code, OpenAI Codex, Grok, Gemini, LLM evaluation, guardrails, model routing, self-hosted open-weight models.

Languages: Python, PHP, TypeScript, JavaScript, SQL, Bash. Infrastructure: Linux, Docker, systemd, nginx, Caddy, Cloudflare, Tailscale, MySQL/MariaDB.

Education

B.S. Computer Science, Business Administration minor, Southern Oregon University, 2010